Using Quantified Epistemic Logic as a Modeling Tool in Cognitive Neuropsychology

Rasmus K. Rendsvig
Philosophy and Science Studies, Roskilde University

1 Introduction

The classic *modus operandi* for model construction in cognitive neuropsychology (CN) is by use of box-and-arrow diagrams[1] to capture the functional architecture of the mental information-processing system under consideration. In such diagrams, of which Figure 1 is an example, boxes represent particular components of the system and arrows represent pathways of information flow. Along with a suitable interpretation, box-and-arrow diagrams represent typical CN theories, consisting of statements regarding what specific modules are included in the system as well as statements regarding how information may flow between these components.

Recently, it has been argued that this methodology should be augmented by the use of computational models, allowing for the realization of CN theories in the form of executable computer programs, structurally isomorphic to the box-and-arrow version the theory, [3, p. 166]:

> This way of doing cognitive psychology is called computational cognitive psychology, and its virtues are sufficiently extensive that one might argue that all theorizing in cognitive psychology should be accompanied by computational modeling.

Though *epistemic logics* are not by themselves executable programs, as a modeling tool they do however possess many of the features and virtues required by [3]. It is the purpose of this paper to discuss the viability of using epistemic logics as a modeling tool in cognitive neuropsychology.

Benefits of epistemic logic. There are three fields that could benefit from epistemic logical modeling of theories from cognitive neuropsychology. First, a motivation for constructing formal logical models aimed at cognitive neuropsychology is that precision and logical entailment can provide explanatory force and working hypotheses. Further, epistemic logics can be used to express higher cognitive functions such as knowledge and belief in straightforward languages. This further allows the formulation of derived principles, such as object recognition. That epistemic logics can straightforwardly express such functions further makes it easy to read off predictions from logical theories, allowing for simple comparisons with empirical observations.

Second, logicians interested in modeling information flow and communication acts could gain more realistic models of the internal parts of these processes, if they took to modeling the functional architecture underlying our abilities to perform such actions.

[1] The "universal notation in modern cognitive neuropsychology", according to [4].

Finally, if logicians and cognitive neuropsychologists merged formal tools and empirical insight, philosophers would stand to gain. Having empirical theories couched in flexible modal logical frameworks would allow precise analyses of philosophical problems using tools with which many philosophers are already familiar. To exemplify, the model introduced below has been used in [14] to give novel analyses of Frege's Puzzle about Identity, based on formal notions of semantic competence.

Plan of attack. As argued in [3], a proper modeling of a CN theory should be true to that theory in an isomorphic sense: the formal model should include exactly the modules and pathways of the CN theory, while maintaining their separate and joint functions. However, while many CN theories include a *semantic system* module, it is far from clear where such a collection of concepts can be found in, e.g., a quantified $\mathsf{S5}$ logic. To overcome this difficulty, we here take the perspective that a logical model of a CN theory is composed of *both* a formal logic as well as semantics for this logic. More specifically, then the modules of a functional architecture is represented by model-theoretic structures over which agents' capabilities can then be expressed using formal logical syntax. A complete logic for the model-theoretic structure may then be seen as a theory detailing the capabilities of an agent with such a mental makeup.

The present approach differs from the computational cognitive neuropsychological (CCN) way outlined in [3] in an important aspect. The CCN approach there discussed construct a model of normal behavior, which can the be 'lesioned' to simulate brain damage, and thereby compare to experimental observations. In the present, we reverse this approach. Instead of constructing first a stronger logic for normal behavior from which we can then remove, we construct a weaker logic for the completely damaged, to which abilities can then be added. As such, the logic is generic: in order to produce a subject-specific logic, further assumptions regarding specific knowledge and abilities must be made.

Evaluating logical models. Given that a motivation for logical modeling of CN theories was a promise of working hypotheses, one would expect such hypotheses to be empirically testable. As the observables in CN tests are subjects' performance based on given input, and these abilities are described by formal logical statements over the model-theoretic structure modeling the functional architecture, it is thus natural to take such hypotheses as constituted by the formulas satisfied in the actual world of the semantic model. As will be shown below, this is a feasible way of comparing formal model with empirical observations. However, there is no hope that *any* normal epistemic logical model can produce hypotheses all of which are consistent with made observations. *Qua* the problem of logical omniscience, any modeled agent will always know all logical consequences of her knowledge [7], which *no* subject can do. This results in a problem for the present approach in relation to evaluating, rejecting and refining the formal models, for every hypothesis may now be wrong for two reasons: it may be the case that the modeled functional architecture does not represent reality, or it may be that the chosen hypothesis requires reasoning skills beyond normal, human capabilities. In the latter case, this will result in a non-answer, and the danger is that the correct functional architecture is rejected. We offer no solution here, but note that this is a problem for which one needs to control when performing theory evaluation.

Structure. The structure of the paper is as follows. In the ensuing section, we introduce a simplified version of a CN theory of the structure of lexical, semantic competence (SLC) from [12]. This is to act as a toy CN theory, which will be modeled using a two-sorted quantified epistemic logic (QEL) introduced in section 3. Most weight is on section 4, where the connections between the QEL and the SLC are drawn. It is first argued that the modules of the SLC are represented in the two-sorted model-theory. Secondly, it is argued that the three distinct competence types of the SLC are expressible in the formal language, and that their dissociations are preserved in the two-sorted logic.[2] Finally, we consider studies which show the shortcomings of the present model, and thereby falsify the presented model. We then conclude.

2 The Structure of Lexical Competence

In [12], a box-and-arrow theory of the *s*tructure of semantic, *l*exical *c*ompetence (SLC) is constructed on the basis of studies from cognitive neuropsychology.[3] The elements of the SLC consist of three competence types defined over four ontologies, two of these being mental modules, see Figure 1. The three competence types are *inferential competence* and two types of *referential competence*, being *naming* and *application*. The four ontologies include one of *external objects*, one of *external words* (e.g., spoken or written words) and two mental modules: a *word lexicon* and the *semantic system*. This structure is illustrated in Figure 1.

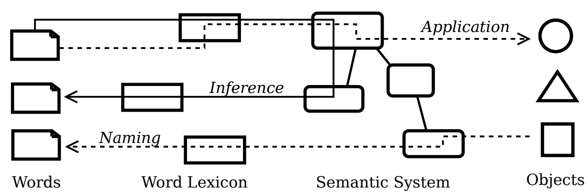

Fig. 1. A simplified illustration of the SLC. Elements in the WL are not connected, only elements in the SS are. Inferential competence requires connecting two items from the word lexicon *through* the semantic lexicon.

Word Lexicon, Semantic Lexicon and Inferential Competence. *Inferential competence* lies between three ontologies: *external word*, *word lexicon* and the *semantic system*. On input, the external word is first analyzed and related to an mental representation from the word lexicon (WL). In [12], two word lexica are included for different input, a phonetic and a graphical. Here, attention is restricted to a simplified structure with only one arbitrary such, consisting only

[2] Due to space restrictions, proof theory will not be considered, but a complete axiomatization can easily be constructed based on the general completeness result for many-sorted modal logics from [13].

[3] For the review of these studies, arguments for the structure and references to relevant literature, the reader is referred to [12]. The presentation here differs slightly, but in-essentially, from that.

of proper names,[4] as illustrated in Figure 1. Using a graphical lexicon as an example, the word lexicon consists of the words an agent is able to recognize in writing. Secondly, the mental representation of the word is related to a mental concept in the semantic system (SS). The SS is a collection of non-linguistic, mental concepts possessed by the agent, *distinct* from the WL. The semantic system reflects the agent's mental model of the world, and the items in this system stand in various relations to one another.[5] In contrast, in the WL connections between the various items *do not exist*. Such only exist *via* the SS. The third step is exactly a connection between two entries in the SS. Finally, the latter of these are connected to an entry in the WL and output can be performed.[6] *Inferential competence* is the ability to correctly connect lexical items via the SS, e.g., connecting 'dog' to 'animal'. This ability underlies performance such as stating definitions, paraphrasing and finding synonyms.

Referential Competence and External Objects. *Referential competence* is "the ability to map lexical items onto the world" [12, p. 60]. This is an ability involving all four ontologies, the last being *external objects*. It consists of two distinct subsystems. The first is *naming.* This is the act of retrieving a lexical item from the WL when presented with an object. It is a two-step process, where first the external object is connected to an suitable concept in the SS, which is then connected to a WL item for output. The second subsystem is that of *application.* Application is the act of identifying an object when presented with a word. Again, this is a two-stage process, where first the WL item is connected to an SS item, which is then connected to an external object. A naming or application deficit can occur if either stage is affected: if, e.g., either an object is not mapped to a suitable concept due to lack of recognition, or a suitable concept is not mapped to the correct (or any) word, then a naming procedure will not be successfully completed.

Empirical Backing for Multiple Modules and Competence Types. The SLC may seem overly complex. It may be questioned why one should distinguish between word and semantic type modules, or why referential competence is composed of two separate competence types, instead of one bi-directional. The reasons for these distinctions are based on empirical studies from cognitive neuropsychology where reviews of subjects with various brain-injuries indicate that these modules of human cognition are separate (see [12] for references).

The distinction between WL and SS is further supported in [8] by cases where patients are able to recognize various objects, but are unable to name them (they cannot access the WL from the SS). In the opposite direction, cases are reported where patients are able to reason about objects and their relations when shown

[4] To only include proper names is technically motivated, as the modeling would otherwise require second-order expressivity. This is returned to below.

[5] Marconi uses the term '*semantic lexicon*', but to keep this presentation in line with standard cognitive neuropsychological terminology, 'semantic system' is used instead.

[6] For simplicity, a distinction between input and output lexica will not be made. See [15] for discussion.

objects, yet unable to do the same when prompted by their names (i.e., the patients cannot access the SS from the WL). The latter indicates that reasoning is done with elements from the SS, rather than with items the WL.

Regarding competence types, it is stressed in [12] that inferential and referential competence are distinct abilities. Specifically, it is argued that the ability to name objects does not imply inferential competence with the used name, and, *vice versa*, that inferential knowledge about a name does not imply the ability to use it in naming tasks. No conclusions are drawn with respect to the relationship between inferential competence and application. Further, application is dissociated from naming, in the sense that application can be preserved while naming is lost. No evidence is presented for the opposite dissociation, i.e. that application can be lost, but naming maintained.

In the following section, a model will be constructed which include the mentioned ontologies over which the competence types can be defined, and in which these are appropriately dissociated.

3 Modeling the Structure of Lexical Competence

To construct a toy model of the SLC, a two-sorted first-order epistemic logic will be used. A very limited syntax is used, though the syntax and semantics could easily be extended to include more agents, sorts, function- and relation symbols, cf. [13].

A two-sorted language is used to ensure that the model respects the dissociation of word lexicon and semantic system. The first sort, σ_{OBJ}, is used to represent external objects and the semantic system entries. As such, these are *non-linguistic* in nature. The second sort, σ_{LEX}, is used to represent the external words from the agent's language and entries in the word lexicon. Had terms been used to represent both simultaneously, the model would be in contradiction with empirical evidence.

The choice of quantified epistemic logic (QEL) fits well with the SLC, if one assumes the competence types to be (perhaps implicitly) knowledge-based.[7] The notions of object identification required for application is well-understood as modeled in the quantified S5 framework, cf. [5]. The 'knowing who/what' constructions using *de re*-constructions in QEL from [10] captures nicely the knowledge required for object identification by the subjects reviewed in [12]. This is returned to in the following section.

Syntax. Define language $\mathcal{L}$ to include two sorts, σ_{OBJ} and σ_{LEX}. For sort σ_{OBJ}, include 1) a countable set of *object constant symbols*, $OBJ = \{a, b, c, ...\}$, and 2) a countably infinite set of *object variables* $VAR = \{x_1, x_2, ...\}$. The set of terms of sort σ_{OBJ} is $TER_{OBJ} = OBJ \cup VAR$. For sort σ_{LEX}, include 1) a countable set of *name constant symbols*, $LEX = \{n_1, n_2, ...\}$, and 2) a countably infinite set of *name variables*, $VAR_{LEX} = \{\dot{x}_1, \dot{x}_2, ...\}$. The set of terms of sort σ_{LEX} is $TER_{LEX} = LEX \cup VAR_{LEX}$.

[7] In [3], visual recognition tasks are explicitly referred to in terms of knowledge (see, e.g., p. 149). Which type of knowledge is however not discussed.

Include further in $\mathcal{L}$ a unary function symbol, μ, of arity $TER_{LEX} \longrightarrow TER_{OBJ}$. The set of all *terms, TER,* of $\mathcal{L}$ are $OBJ \cup VAR \cup LEX \cup VAR_{LEX} \cup \{\mu(t)\}$, for all $t \in LEX \cup VAR_{LEX}$. Finally, include the binary relation symbol for identity, $=$. The well-formed formulas of $\mathcal{L}$ are given by

$$\varphi ::= (t_1 = t_2) \,|\, \neg\varphi \,|\, \varphi \wedge \psi \,|\, \forall x\varphi \,|\, K_i\varphi$$

The definitions of the remaining boolean connectives, the dual operator of K_i, $\hat{K}_i$, the existential quantifier and free/bound variables and sentences are all defined as usual. Though a mono-agent system, the operators are indexed by i to allow third-person reference to agent i.

Semantics. Define a 2QEL *model* to be a quadruple $M = \langle W, \sim, Dom, \mathcal{I} \rangle$ where

1. $W = \{w, w_1, w_2, ...\}$ is a set of *epistemic alternatives* to actual world w.
2. $\sim$ is an *indistinguishability (equivalence) relation* on $W \times W$.
3. $Dom = Obj \cup Nam$ is the *(constant) domain of quantification*, where $Obj = \{d_1, d_2, ...\}$ is a non-empty set of *objects*, and $Nam = \{\dot{n}_1, \dot{n}_2, ..., \dot{n}_k\}$ is a finite, non-empty set of *names*.
4. $\mathcal{I}$ is an *interpretation function* such that
$$\mathcal{I} : OBJ \times W \longrightarrow Obj \,|\, \mathcal{I} : LEX \longrightarrow Nam \,|\, \mathcal{I} : \{\mu\} \times W \longrightarrow Obj^{Nam}$$

To assign values to variables, define a *valuation function*, v, by

$$v : VAR \longrightarrow Obj \,|\, v : VAR_{LEX} \longrightarrow Nam$$

and a *x-variant of* v as a valuation v' such that $v'(y) = v(y)$ for all $y \in VAR_{(LEX)}/\{x\}$.

Truth conditions for formulas of $\mathcal{L}$ are now defined as follows:

$$M, w \models_v (t_1 = t_2) \text{ iff } d_1 = d_2, \text{ where } d_i = \begin{cases} v(t_i) & \text{if } t_i \in VAR \cup VAR_{LEX} \\ \mathcal{I}(w, t_i) & \text{if } t_i \in OBJ \\ \mathcal{I}(t_i) & \text{if } t_i \in LEX \end{cases}$$

$$\begin{array}{ll} M, w \models_v \varphi \wedge \psi & \text{iff } M, w \models_v \varphi \text{ and } M, w \models_v \psi \\ M, w \models_v \neg\varphi & \text{iff not } M, w \models_v \varphi \\ M, w \models_v K_i\varphi & \text{iff for all } w' \text{ such that } w \sim w',\ M, w' \models_v \varphi \\ M, w \models_v \forall x\varphi(x) & \text{iff for all } x\text{-variants } v' \text{ of } v,\ M, w \models_{v'} \varphi(x) \end{array}$$

Comments on the semantics are postponed to the ensuing section.

Logic. A sound and complete two-sorted logic for the presented semantics can be found in [13]. The logic is here denoted $\mathsf{QS5}_{(\sigma_{LEX}, \sigma_{OBJ})}$.

4 Model Validation

As mentioned above, the modules of the functional architecture of the SLC is represented by model-theoretic structures, over which the agent's capabilities can then be expressed using the formal logical syntax. So far, the structures introduced bear little resemblance to the SLC, and it will be a first task to extract this hidden structure. Secondly, it is shown that the logical model can express the three competence types and that the dissociation properties are preserved in the logic.

4.1 Ontologies

The two sets of external objects and external words are easy to identify in the model-theoretic structure. The external objects constitute the sub-domain Obj, and are denoted in the syntax by the terms TER_{OBJ}, when these occur outside the scope of an operator. External words (proper names) constitute the sub-domain Nam denoted by the terms TER_{LEX}, when occurring outside the scope of an operator.

The word lexicon and the semantic system are harder to identify. The strategy is to extract a suitable notion from the already defined semantic structure. To bite the bullet, we commence with the more complicated semantic system.

Semantic System. In order to include a befitting, albeit very simple notion, define an *object indistinguishability relation* $\sim^a_w \subseteq Obj \times Obj$:

$$d \sim^a_w d' \text{ iff } \exists w' \sim w : \mathcal{I}(a, w) = d \text{ and } \mathcal{I}(a, w') = d'.$$

and from this define the agent's *individual concept class for* a *at* w by $C^a_w(d) = \{d' : d \sim^{a,w}_i d'\}$. The semantic system of agent i may then be defined as the collection of non-empty concept classes: $\mathsf{SS}_i = \{C^a_w(d) : C^a_w(d) \neq \emptyset\}$.

The set $C^a_w(d)$ consists of the objects indistinguishable to the agent by $a \in OBJ$ from object $d \in Obj$ in the part of the given model connected to w by $\sim$. As an example, consider a scenario with two cups (d and d' from Obj) upside down on a table, where one cup conceals a ball. Let a denote *the cup containing the ball*, say d, so $\mathcal{I}(a, w) = d$. If the agent is not informed of which of the two cups contains the ball, i.e. which is a, there will be an alternative w' to w such that $\mathcal{I}(a, w') = d'$. Hence, $d \sim^a_w d'$ so $d' \in C^a_w(d)$. The interpretation is that the agent cannot tell cups d and d' apart *with respect to which conceals the ball.*[8]

Important properties of individual concepts can be expressed in $\mathcal{L}$ (see [13]). For present purposes, most importantly we have that

$$|C^a_w(d)| = 1 \text{ iff } M, w \models_v \exists x K_i(x = a), \tag{1}$$

i.e. the agents has a singleton concept of a in w iff it is the case that the agent *knows which object* a *is*, in the readings of [9,5]. The intuition behind this reading is that the satisfaction of $\exists x K_i(x = a)$ requires that the interpretation of a is constant across i's epistemic alternatives. Hence, there is no uncertainty for i with respect to which object possesses feature a – i knows which object a is. Using a contingent identity system for objects, i.e. giving these a non-rigid interpretation as done in the semantics above, results in the invalidity of both $(a = b) \to K_i(a = b)$ and $(a = b) \to \exists x K_i(x = b)$. Hence, agent i does *not* by default know whether objects are identical when identified by different features, and neither is the agent able to *identify* objects by default – as in the example above. This is a good example of how the present is a weak, generic model: subject specific abilities regarding identificatory abilities needs to be made as further assumptions on a per subject basis.

Word Lexicon. A suitable representation of the word lexicon is simpler to extract than for the SS. This is due to the non-world relative interpretation of

[8] Though the agent may be able to tell them apart with respect to other features, like their color or position.

name constants $n \in LEX$, which so far has gone without comment. The interpretation function $\mathcal{I}$ of the name constants is defined constant in order ensure that the agent is *syntactically competent*. From the definition of $\mathcal{I}$, it follows that $(n_1 = n_2) \rightarrow K_i(n_1 = n_2)$ is valid on this class of models. This corresponds formally to the incontingent identity system used in [11]. The interpretation is that whenever the agent is presented by two name tokens of the same type of name, the agent knows that these are tokens of the same name type. The assumptions is adopted as the patients reviewed in [12] are able to recognize the words utilized.

Notice that identity statements such as $(n_1 = n_2)$ *do not* convey any information regarding the *meaning* of the names. Rather, they express identity of the two *signs*. Hence, the identity '*London* = *London*' is true, where as the identity '*London* = *Londres*' is false – as the two first occurrences of '*London*' are two tokens (e.g. $n_1, n_2 \in LEX$) of the same type (the type being $\dot{n} \in Nam$), whereas the '*London*' and '*Londres*' are occurrences of two different name types, albeit with the same meaning.

Due to the simpler definition of $\mathcal{I}$ for name constants, we can define i's *name class for* n directly. Where $\dot{n} \in Nam$ and $n \in LEX$ this is the set $C_i^n(\dot{n}) = \{\dot{n'} : \mathcal{I}(n) = \dot{n}'\}$. The word lexicon of i is then the collection of such sets: $\mathsf{WL}_i = \{C_i^n(\dot{n}) : n \in LEX\}$. Each name class is a singleton equivalence class, and WL_i is a partition of Nam. Further, (1) holds for name classes if suitably modified, and the construction of WL_i therefore fits nicely fit the assumption of syntactic competence.

4.2 Interlude: Word Meanings

In order to model knowledge of the meanings of word tokens $n \in LEX$, these must first be assigned a meaning. In the clinical trials reviewed in [12], applying a name to it's meaning is done by extension identification. Therefore, a simple purely extensional theory of meaning have been embedded in the framework: the function symbol μ of arity $TER_{LEX} \longrightarrow TER_{OBJ}$. A meaning function rather than a relation is used as only proper names are included in the agent's language, and for these to have unambiguous meanings, the function requirement is natural. Given it's defined arity, μ assigns an element of TER_{OBJ} to each name in TER_{LEX}. From the viewpoints of the agents, μ hence assigns an object (meaning) to each name.

On the semantic level, $M, w \models_v (\mu(n) = a)$ is taken to state that the meaning of name n is the object a in the actual world w. The reference map μ is defined *world relatively*, i.e. the value $\mu(n)$ for $n \in LEX$, can change from world to world. This is the result of the world relative interpretation of μ given in the semantics above. Hence, *names are assigned values relative to epistemic alternatives*.

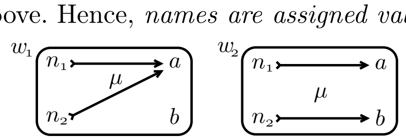

Fig. 2. The meaning function μ is defined world relatively, so meaning of a name may shift across epistemic alternatives.

The motivation for a world relative meaning function is the generic nature of the model. No agents will *by default* have knowledge of the meaning of words from their language, but further assumptions to that effect can be assumed in specific cases.

The simplifying assumption that the `WL` should include only proper names was technically motivated by the inclusion of μ. Had verbs been introduced in the agent's language, then μ should have assigned them relation symbols as meanings, and a second-order logic would be required.

4.3 Competence Types

Inferential Competence. Due to the restriction to proper names, the model is extremely limited in the features expressible regarding inferential competence. The expressible instances of inferential competence are limited to knowing relations between referring names and not inferential knowledge regarding names and verbs. As an example, one cannot express that the agent knows the true sentence 'name is verb' as the word lexicon does not contain a 'verb' entry. We are however able to express *knowledge of co-reference*:

$$K_i(\mu(n) = \mu(n')) \tag{2}$$

(2) states that i knows that n and n' co-refer, i.e. knows the two names to be synonyms. Based on (2), we may define that agent i is *generally inferentially competent with respect to* n by

$$M, w \models_v \forall \dot{x}((\mu(n) = \mu(\dot{x})) \to K_i(\mu(n) = \mu(\dot{x}))) \tag{3}$$

where $\dot{x} \in VAR_{LEX}$. If (3) is satisfied for all n, agent i will have full 'encyclopedic' knowledge of the singular terms of her language. This may however be 'Chinese Room style' knowledge, as it does not imply that any names can be applied nor that any objects can be named.

Referential Competence. Referential competence compromises two distinct relations between names and objects, relating these through the semantic system, namely *application* and *naming*. An agent can *apply a name* if when presented with a name, the agent can identify the appropriate referent. This ability can be expressed of agent i with respect to name n in w by

$$M, w \models_v \exists x K_i(\mu(n) = x) \tag{4}$$

i.e. there is an object which the agent can identify as being the referent of n. Given the assumption of syntactical competence, there is no uncertainty regarding which name is presented. Since the existential quantifier has scope over the knowledge operator, the interpretation of $\mu(n)$ is fixed across epistemic alternatives, and i thus knows which object n refers to.

To be able to *name an object*, the agent is required to be able to produce a correct name when presented with an object, say a. For this purpose, the *de re* formula $\exists \dot{x} K_i(\mu(\dot{x}) = a)$ is insufficient as $\mu(\dot{x})$ and a may simply co-vary across states. This means that i will be guessing *about which object is to be named*, and may therefore answer incorrectly. Since there may in this way be uncertainty regarding presented objects, naming must include a requirement that i can identify a, as well as know a name for a. This is captured by

$$M, w \models_v \exists x \exists \dot{x} K_i((x = a) \wedge (\mu(\dot{x}) = a)). \tag{5}$$

Here, the quantification and first conjunction ensures that i can identify the presented object a and the second conjunct ensures that the name refers to a in all epistemic alternatives.

Dissociations. As mentioned, inferential competence and naming are dissociated. This is preserved in the model in that neither (2) nor (3) alone imply (5). Nor does (5) alone imply either of the two. The dissociation of application from naming is also preserved, as (4) does not alone entail (5). That application does not imply naming is illustrated in Figure 3.

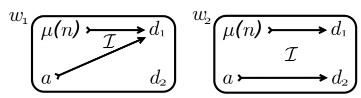

Fig. 3. Application and naming are not correlated. In actual world w_1, n refers to a and i can correctly apply n, but cannot name a using n: $w_1 \models_v (\mu(n) = a) \wedge \exists x K_i(\mu(n) = x)$, but $w_1 \models_v \neg\exists x \exists \dot{x} K_i((x = a) \wedge (\mu(\dot{x}) = a))$. Here, i cannot name a due to an ambiguous concept. a may be either of d_1 or d_2, and can therefore not be identified precisely enough to ensure a correct answer.

Whether application entails inferential competence, and whether naming entails application is not discussed in [12]. In the present model, however, these are modeled as dissociated in the sense that (4) does not entail, nor is entailed by, either (3) or (5). However, the modeled dissociations are *single instances* of the various abilities. Once more instances are regarded simultaneously, implicational relationships arise, as will be discussed below.

5 Hypotheses and Explanations

A motivation for constructing formal logical models is that precision and logical entailment can provide explanatory force and working hypotheses. One testable hypothesis of the present model predicts *lack of dissociation* between multiple application instantiations and inferential competence. Specifically, the model entails that subjects capable of applying two co-referring names will be knowledgeable of their co-reference:

$$\begin{gathered}\text{If } M, w \models_v (\mu(n) = \mu(n')) \text{ then} \\ M, w \models_v \exists x K_i(\mu(n) = x) \wedge \exists y K_i(\mu(n') = y) \to K_i(\mu(n) = \mu(n'))\end{gathered} \tag{6}$$

From this, an explanation why none of the studies reviewed in [12] show dissociation between application and inferential competence can be conjectured: simple inferential competence can come about as a bi-product of application, memory and deductive skill and may therefore require much damage before being severely impaired.

The formalizations of application and naming also suggests a reason why no cases where naming was intact, but application broken, was reported in [12]: the ability to name is very close to entailing the ability to apply a name. In fact, once (5) is instantiated with a specific name, it implies (4) for the same name. For application, the chosen object is identified by the subject *via* the mental representation 'the referent of n', whereas for naming, the presented object must first be identified by some other feature, e.g., a visual trademark. In case the mental representation in the `SS` of this feature is then identical to that of 'the referent of n', then the subject will be able to name a. Hence, one of the necessary conditions for naming almost implies the necessary and sufficient condition for application, why the latter will be observed accompanying the former.

Implicational relationships as the mentioned should allow for the refutation of the model. If subjects are found who possess abilities represented by the antecedents, but lack those of the consequents, the model can be regarded as refuted, though exactly what the problem is may not be obvious, qua the previously mentioned issue with logical omniscience.

As mentioned above, both the difference between orthographic and and phonological lexica as well as the difference between input and output lexica was ignored in this presentation. Since the logical model therefore is based on an arguably wrong functional architecture, it should be possible to find inconsistencies between model and observed subject behavior. This is indeed the case. For, if the presented model was correct, then the word lexicon entries should play both orthographic and phonological roles for words the agent knows in both speech and writing. Given such a word, an agent able to name with the word should always be able to do so both orally and in writing. That is, the hypothesis that agent i is able to name a, i.e. $\exists x \exists \dot{x} K_i((x = a) \land (\mu(\dot{x}) = a))$ ((5) from above), requires that the subject can produce name(s) $\dot{x}$ both orally and in writing. This, however, is not the case, as is illustrated, e.g., by the case of RCM, an 82-year old woman, reported on in [8, p. 191]. When prompted with a picture of a turtle, RCM was able to correctly name it orally using 'turtle', but named it incorrectly in writing, using 'snake'. As RCM repeatedly made similar mistakes with respect to written word tasks but not with oral naming tasks, this case can be taken to show that damage to the orthographic (output) lexicon does not imply damage to the phonological (output) lexicon. This is not possible in the presented model, why the model is refuted.[9]

6 Conclusions and Further Perspectives

In the present paper, we have looked at the possibilities of using epistemic logic as a modeling tool for cognitive neuropsychological theories by a toy model construction. It has been shown how the functional architecture can be represented using a combination of model theory and formal syntax. It was further shown that the constructed model respected important dissociations, but also how the model could be refuted by suitable empirical evidence contradicting an hypothesis of the model. In conclusion, though the model is incorrect and simplistic, a serious epistemic logical approach to modeling functional architecture theories from cognitive science could possibly be of value. An attempt at making a proper model of a full CN theory would be an obvious next step.

A clear limitation of the presented model is that the formal semantic system lacks content. The limitation to objects only should be lifted as to include various properties and relations as well, and moreover, the representation of objects are black boxed behind constants without a precise interpretation. Before a clear picture such concepts' role can be given, an explicit theory of *object recognition* must be incorporated. It would be interesting to see the effects of formalizing, e.g., *geon theory* of [2] and 'plug it in' in the place of the object constants.

[9] This can easily be remedied by distinguishing between phonological and orthographic word terms, i.e. by the addition of a further word sort.

More structure could also be provided by attempting to incorporate elements from *conceptual spaces theory* [6]. Finally, knowledge operators are too strong in some cases – RCM from above being a case in point. In situations where subjects answer consistently but wrongly, belief operators would be better suited. A range of competence levels could be captured using the various operators from [1].

The style of modeling semantic competence presented here differs from the way the conceptual theory of [12] and other cited literature tend to regard these matters. Here, competence types was defined relative to specific words, and competence judged on a case-by-case basis. Many studies from cognitive neuropsychology base their conclusions on percent-wise correct performance over one or more test batteries and therefore focus on impaired connections between modules. In order to facilitate comparison of formal models and empirical research, the case-by-case methodology must be reconciled with this more general approach.

References

1. Alexandru Baltag and Sonja Smets. A Qualitative Theory of Dynamic Interactive Belief Revision. In *LOFT7*, pages 11–58. Amsterdam University Press, 2008.
2. Irving Biederman. Recent Psychophysical and Neural Research in Shape Recognition. In I. Rentschler, I. Biederman, and N. Osaka, editors, *Object Recognition, Attention, and Action*, chapter 5, pages 71–88. Springer, 2007.
3. Max Coltheart. Cognitive Neuropsychology. In John Wixted, editor, *Methodology in Experimental Psychology*, chapter Cognitive Neuropsychology, pages 139–174. John Wiley & Sons, 2002.
4. Max Coltheart. Cognitive Neuropsychology. *Scholarpedia*, 3(2):3644, 2008.
5. Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Reasoning about Knowledge*. The MIT Press, 1995.
6. Peter Gärdenfors. *Conceptual Spaces*. The MIT Press, 2000.
7. Vincent F. Hendricks. *Mainstream and Formal Epistemology*. Cambridge University Press, 2006.
8. Argye E. Hillis. The Organization of the Lexical System. In Brenda Rapp, editor, *The Handbook of Cognitive Neuropsychology*, pages 185–210. Psychology Press, 2001.
9. Jaakko Hintikka. *Knowledge and Belief: An Introduction to the Logic of the Two Notions*. Cornell University Press. Reprinted in 2005, prepared by V.F. Hendricks and J. Symons, King's College Publications, 1962.
10. Jaakko Hintikka. Different Constructions in Terms of 'Knows'. In Jonathan Dancy and Ernest Sosa, editors, *A Companion to Epistemology*. Wiley-Blackwell, 1994.
11. G.E. Hughes and M.J. Cresswell. *A New Introduction to Modal Logic*. Routledge, 1996.
12. Diego Marconi. *Lexical Competence*. The MIT Press, Cambridge, MA, 1997.
13. Rasmus Kræmmer Rendsvig. Towards a Theory of Semantic Competence. Master's thesis, Roskilde University, http://rudar.ruc.dk/handle/1800/6874, 2011.
14. Rasmus Kræmmer Rendsvig. Modeling Semantic Competence: a Critical Review of Frege's Puzzle about Identity. In Dan Lassiter and Marija Slavkovic, editors, *New Directions in Logic, Language, and Computation*. Springer, 2012.
15. Marie-Josèphe Tainturier and Brenda Rapp. The Spelling Process. In Brenda Rapp, editor, *The Handbook of Cognitive Neuropsychology*, pages 263–290. Psychology Press, 2001.